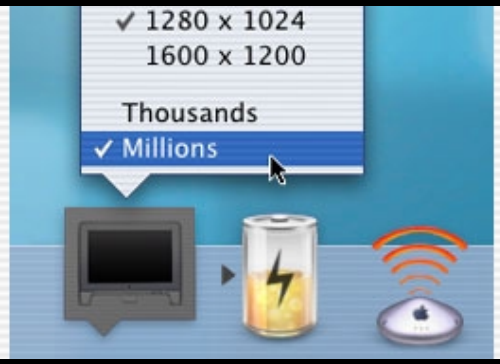
Session 103

Darwin Overview:
The Foundation of Mac OS X



Brett Halle
Director, Core OS Engineering

What Is Darwin?

- Darwin IS the Core OS of Mac OS X
- Open source!

What Is the Core OS?

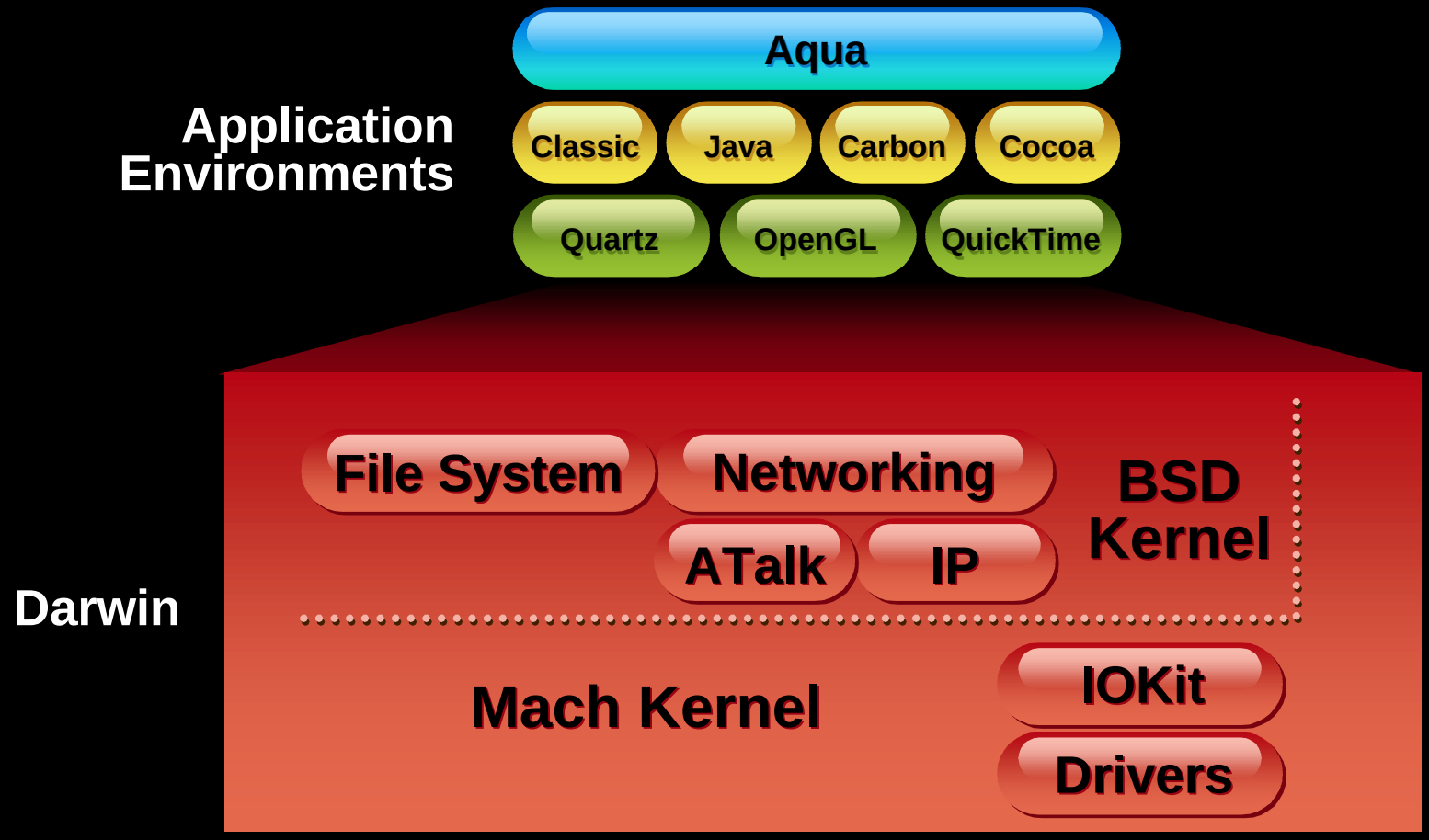
Darwin/Core OS

- Unique architecture
  - Built from multiple best-of-class technologies
- Based on proven technology
  - Hundreds (probably thousands) of man-years invested in development and testing
- Tuned for Apple's customer base
- Lots of untapped potential
  - We are enabling the next killer apps

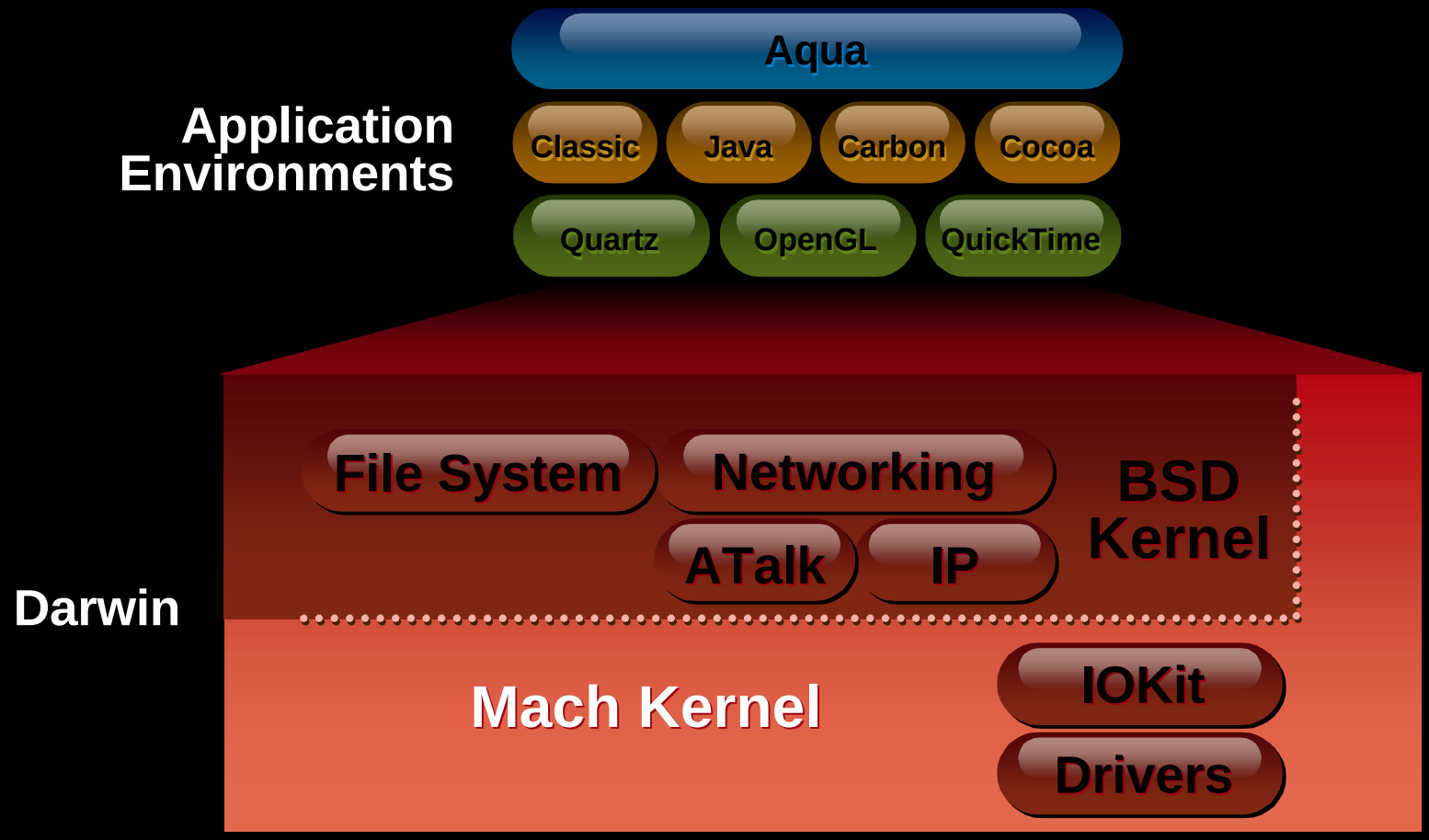
Darwin/Core OS

- Provides preemption and protection
- Multiple address spaces
- Supports the application environments
- Device independent
- Flexible
- Scalable
- Completely buzz-word compliant!

Darwin/Core OS



Mach Kernel


Application Environments
Aqua
Classic
Java
Carbon
Cocoa
Quartz
OpenGL
QuickTime
Darwin
File System
Networking
ATalk
IP
BSD Kernel
Mach Kernel
IOKit
Drivers

Mach: OS Foundation

- Manages processor and memory resources
  - Processor scheduling
    - Threading is an integral kernel service
  - Memory protection, memory management
- Message based architecture (IPC/RPC)
- Policy neutral
- Mach does not provide:
  - I/O, Networking, File Systems, Security Policy, Application APIs, etc.

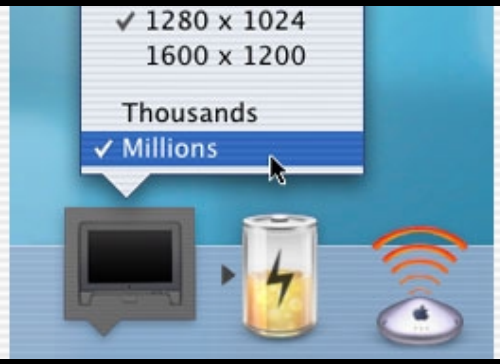
Mach

- Mac OS X: Based on Mach 3.0+
  - Derived from CMU and OSF/RI Mach 3.0 work
    - Plus tons of work by Apple
  - Architectural support for SMP/real-time
  - Multiple scheduling policies
  - External pager
  - Modular architecture

Application View of Mach

- All process models are built on top of Mach primitives
  - Tasks and Threads
- All memory management is built on top of Mach primitives
  - VMObjects, virtual memory, paging
- Kernel to User and User to User communication is built on top of Mach primitives
  - IPC & RPC, “ports”

Demo



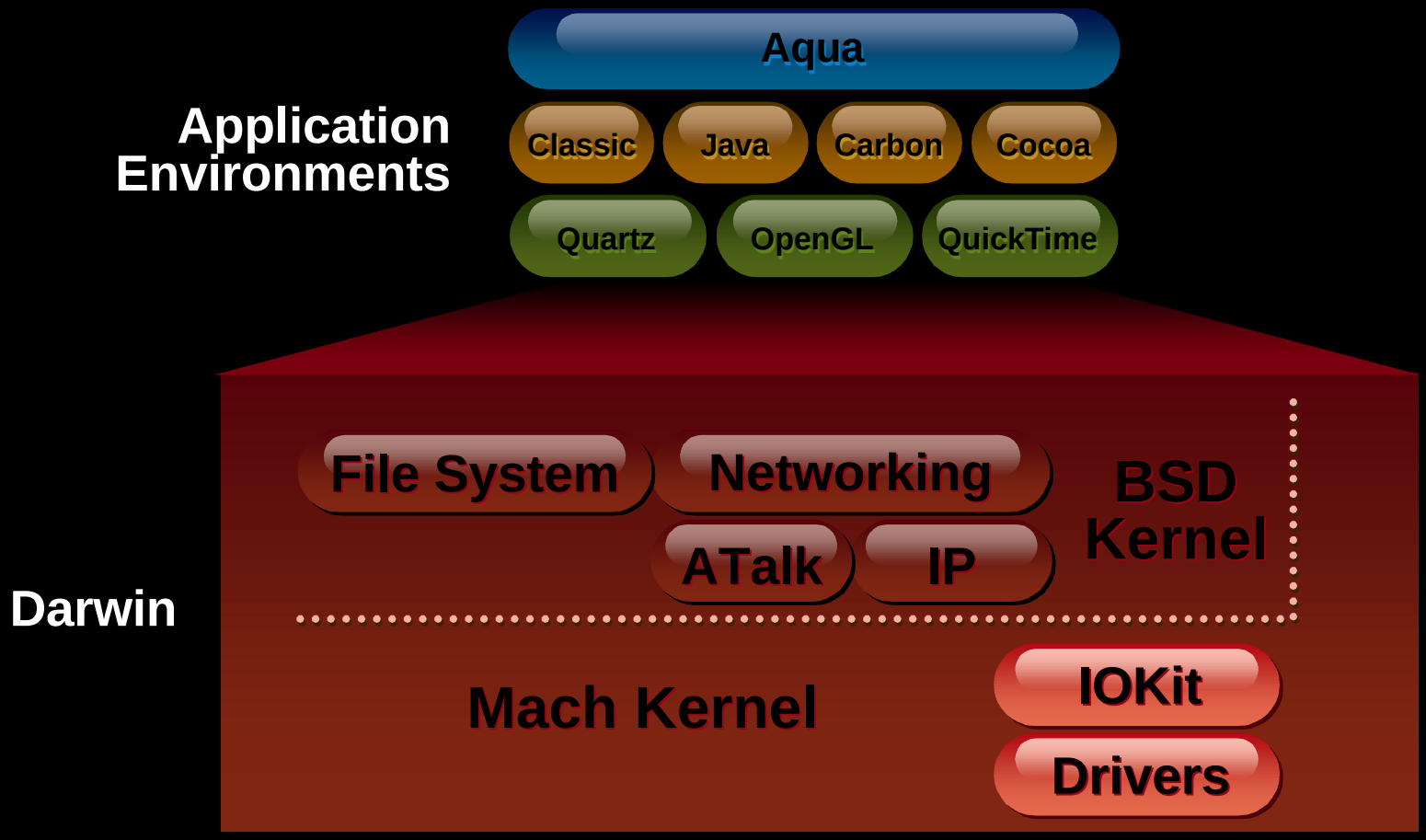
Joseph Sokol, Jr.
Manager, Core OS Performance Group

Roadmap

140 The Darwin Kernel

Civic
Fri., 3:30pm

I/O Kit and Drivers



I/O Kit

- Object-oriented framework for easy development of device drivers
  - True plug-and-play
  - Dynamic device management
  - Power management
  - Modular and extensible
- Provides abstractions for most device classes
  - Mass storage, SCSI, ATA, USB, FireWire, Audio, Network, PCI, etc., etc.

I/O Kit Features

- I/O Kit is a framework
  - Provides common kernel services for I/O
    - User/Client model
  - Models physical world
  - Handles driver matching
- Protocol specifics are in Families
  - For example: SCSI versus Audio
- Device specifics are in Drivers
  - Access to the hardware

Power Management

- Integral to I/O Kit architecture
- Based on the concept of power planes
- Wake up is done in parallel and eliminates as many serial dependencies as possible

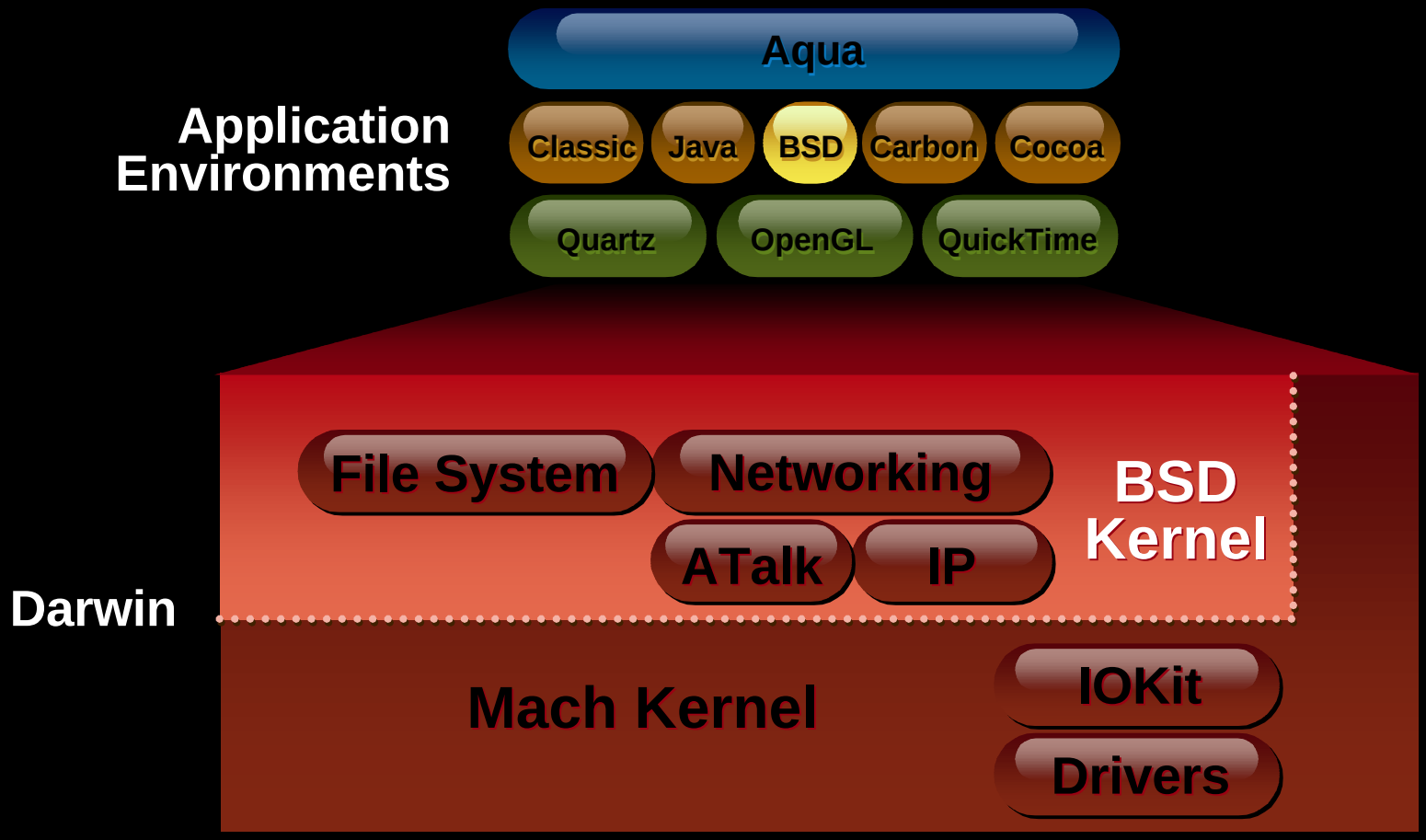
I/O Kit and Applications

- Most devices are hidden behind other OS abstractions
  - Disks: file systems
  - Networking: IP
  - Input devices: events
- User clients exist when applications need device access
  - USB
  - HID Manager

Roadmap

104 I/O Kit Update	Room A1 **Mon., 3:30pm**
202 FireWire Overview	Room A2 **Tues., 2:00pm**
203 USB Overview	Room A2 **Tues., 3:30pm**
204 I/O Kit: Storage Drivers	Room A2 **Tues., 5:00pm**
205 I/O Kit: PCI Drivers and Open Firmware	Room C **Wed., 9:00am**

BSD: The Power of UNIX

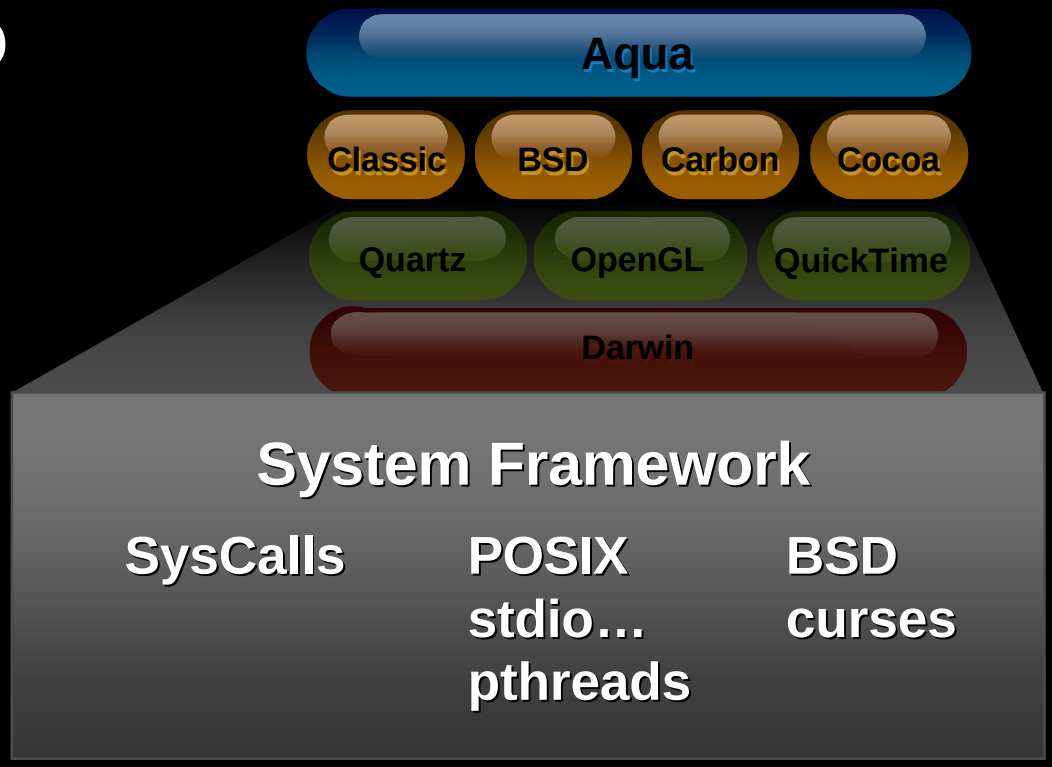


BSD Kernel

- Based on BSD 4.4
  plus integration with Mach and I/O Kit
- Provides “OS Personality” APIs and Services
  - Process model
  - Basic security policy
  - File system architecture
  - Networking architecture

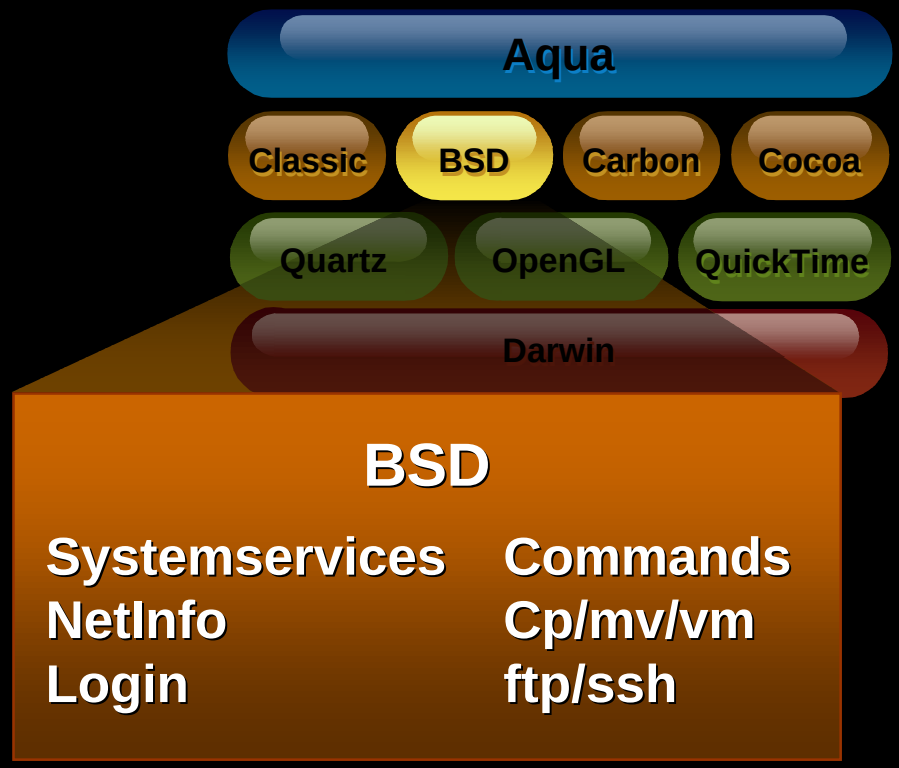
System Framework

- Exposes "POSIX"/BSD System APIs
- Additional services:
  - pthreads
  - Math library
  - C library



BSD User Environment

- Command line/shells
- Network client tools
  - ssh, finger, ftp
- File tools
  - cp, ls, mkdir
- System tools
  - ps, kill, top, shutdown
- System services
  - NetInfo, DNS/BIND, NTP, syslog, Apache



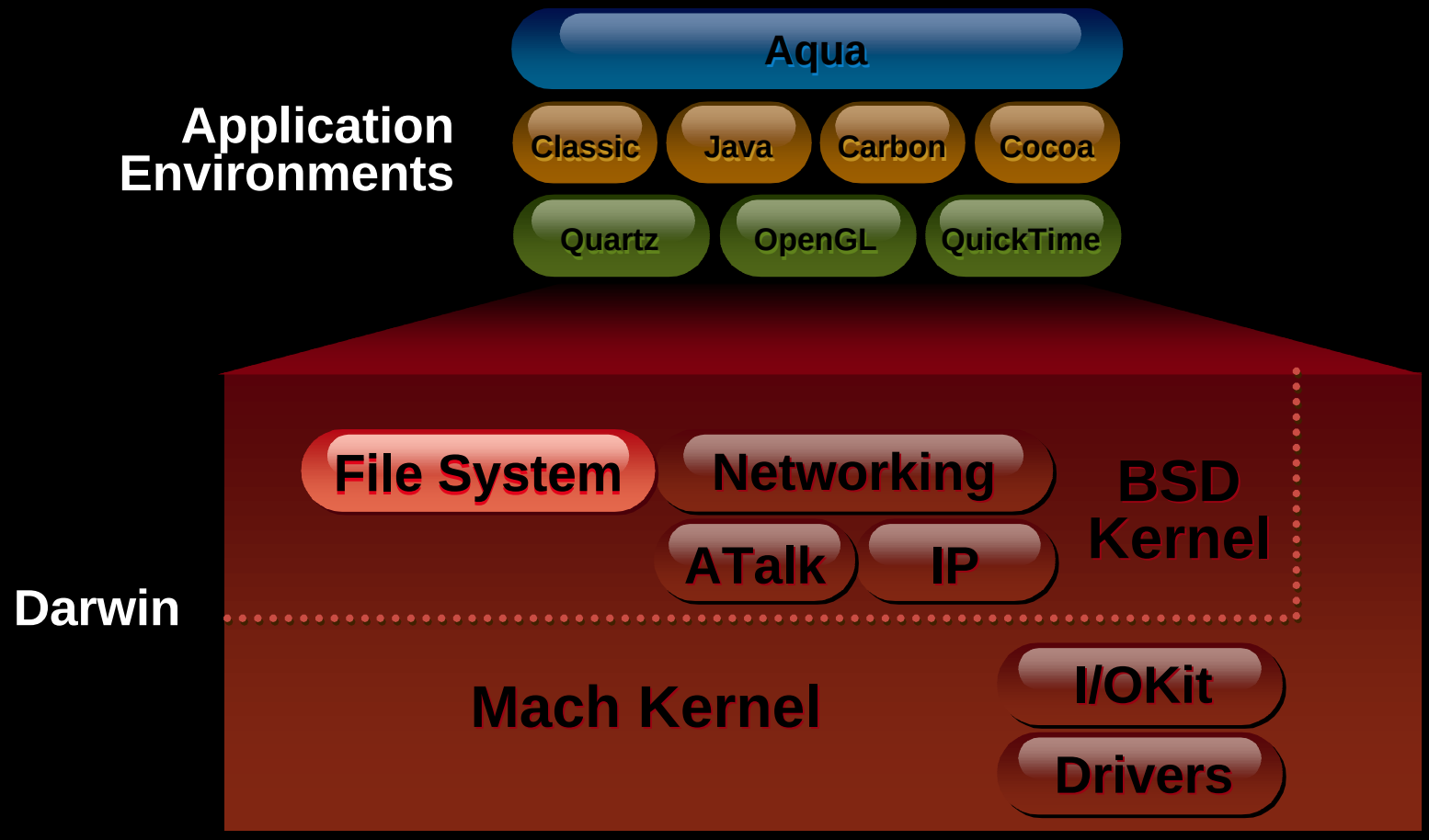
BSD and You

- Process model
  - Environment variables, signals
- Security
  - Users, file system access
- A developers perspective…tools, tools, tools
- Porting UNIX applications to Mac OS X
- Using UNIX tools from within a GUI application

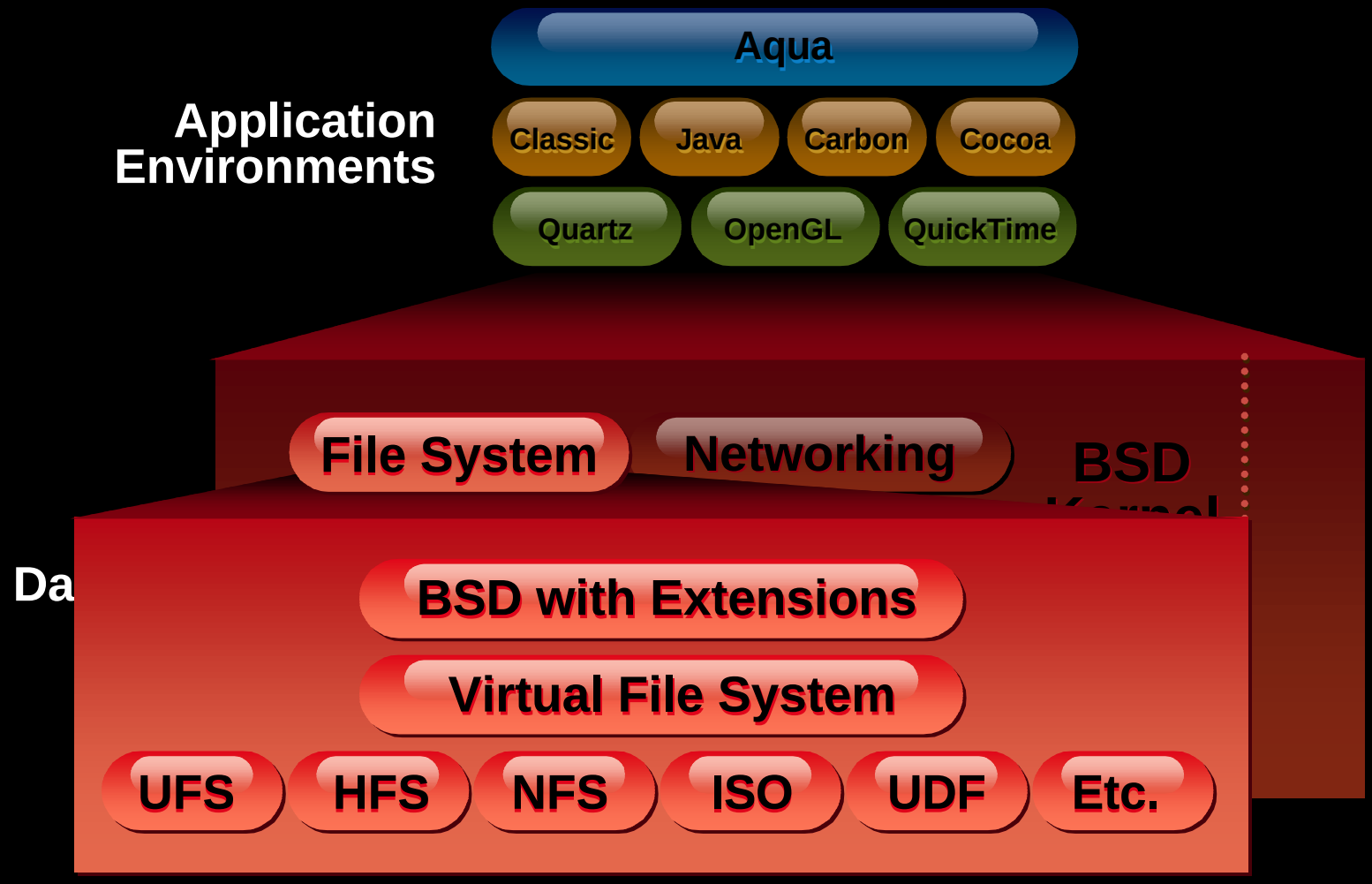
Roadmap

139 Leveraging BSD Services in Mac OS X	Civic **Fri., 2:00pm**
201 How Threading Can Benefit Your Application on Mac OS X	Room A2 **Tues., 9:00am**
302 Directory Services	Room A1 **Tues., 3:30pm**

File System


Application Environments
Aqua
Classic
Java
Carbon
Cocoa
Quartz
OpenGL
QuickTime
Darwin
File System
Networking
ATalk
IP
BSD Kernel
Mach Kernel
I/OKit
Drivers

File System


Aqua
Application Environments
Classic
Java
Carbon
Cocoa
Quartz
OpenGL
QuickTime
File System
Networking
BSD
Da
BSD with Extensions
Virtual File System
UFS
HFS
NFS
ISO
UDF
Etc.

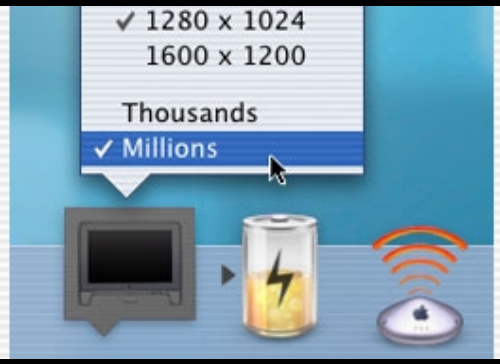
File System

- Component of the BSD kernel environment
- Enhanced VFS design
  - POSIX++ interface
  - Layered architecture (stackable)
- Can support numerous file system types
  - HFS, UFS, ISO 9660, FAT, AFP, NFS, WebDAV, UDF, AFS, etc.
- UTF-8 interfaces
- No specific file system affinity

File System

- Enforces file security policy
  - Users, groups, access control
- Application environments provide abstracted interfaces
  - Carbon: File Manager
  - Cocoa: NSFile
  - BSD: Posix++

Demo



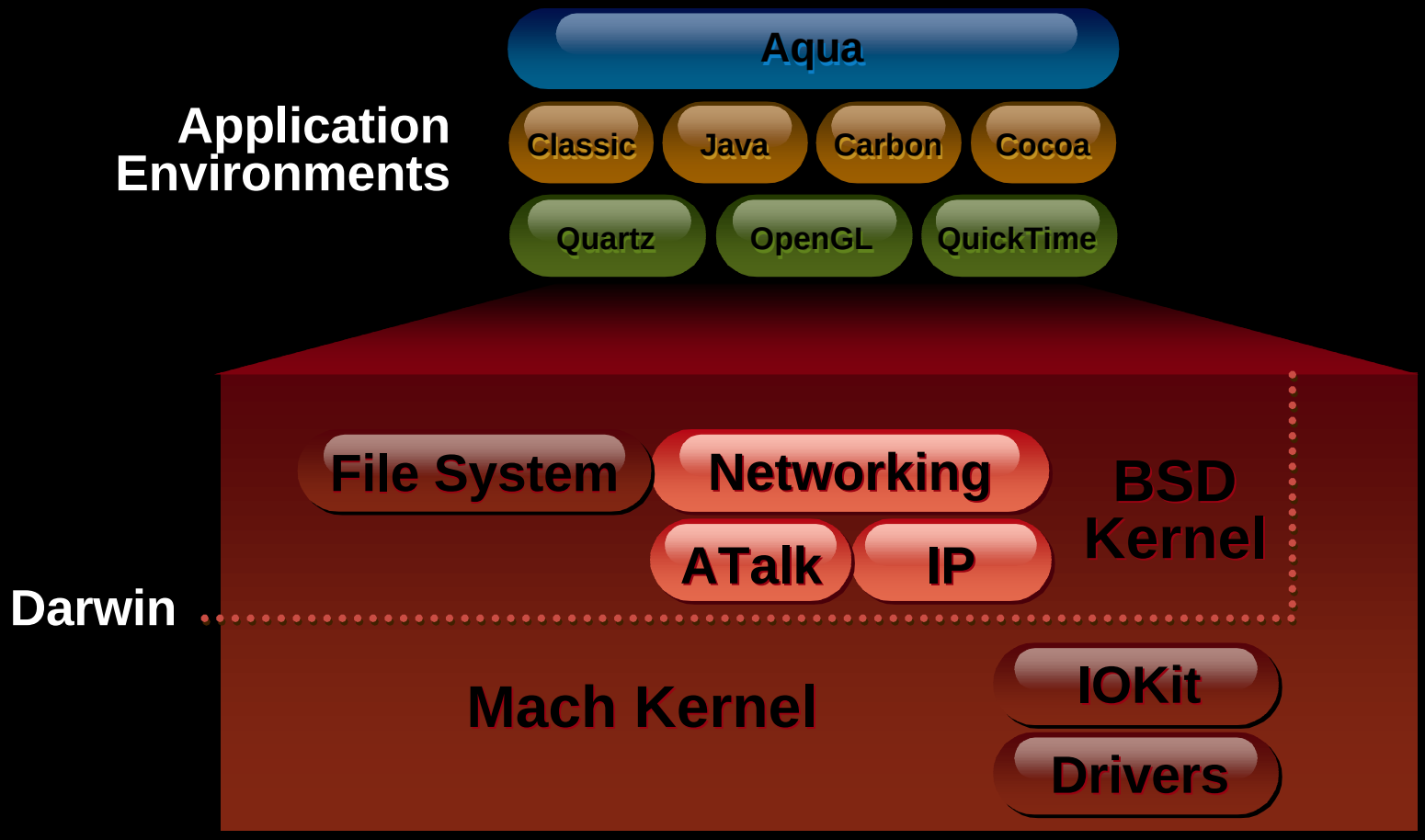
Joseph Sokol, Jr.
Manager, Core OS Performance Group

Roadmap

116 Mac OS X File System

Room J2
Wed., 9:00am

Networking in Mac OS X



Networking

- Component of the BSD kernel environment
- BSD 4.4 TCP/IP Stack (sync'd with FreeBSD 3.2)
  - Socket APIs
- Key features:
  - IPv4 Multi-homing
  - Full IPv4 routing
  - Firewall and NAT mechanisms
- Auto configuration

Networking

- Kernel Extensions for Networking
- Classic support: Shared IP
- Open Transport in Carbon
- PPP, PPPoE
- DHCP client, server
- Ad hoc networks

Mobility

- Flexible architecture
- Auto (re)configuration
  - Link level detection
  - Dynamic
- Multiple configurations
- Application notification

Roadmap

300 Networking Overview	Room A1 **Tues., 9:00am**
304 Extensible Kernel Networking Services	Room J2 **Tues., 10:30am**
303 Network Configuration & Mobility	Room A1 **Tues., 5:00pm**
FF107 Feedback Forum: Networking	Room J **Thurs., 10:30am**

Core OS and G4

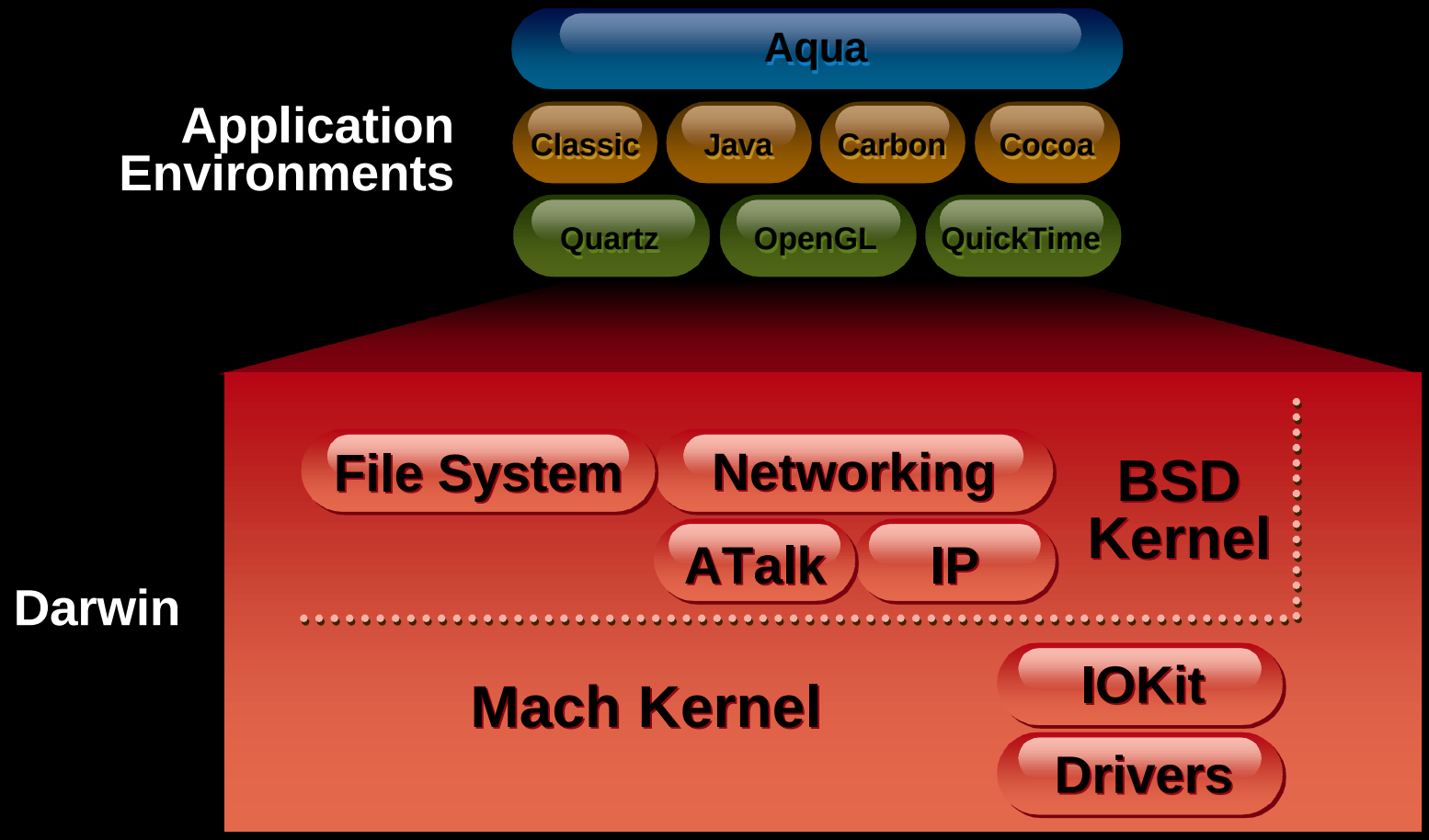
Velocity Engine

- Vector libraries (vecLib framework)
  - vDSP: fast digital signal processing functions, including FFTs and convolutions
    - On Mac OS 9.1 and X
  - vBLAS: vector Basic Linear Algebra Subroutines, including matrix operations, used for MP3, mpeg, speech, image processing
    - Only available on Mac OS X
  - Matrix multiplies on 500mhz G4: 2.2 gigaflops!
  - vMathLib, vectorOps, vBasicOps and vBigNum

Multi-Processor G4's

- Mac OS X supports true SMP
- Each thread can run on a separate processor
  - Use MP APIs for Carbon
  - NSThread for Cocoa
  - pthreads for BSD apps
- Mach, BSD, and SMP…

Kernel Extensions



Kernel Extensions

- A code-fragment container for kernel binaries
  - I/O Kit Drivers
  - Network Kernel Extensions (NKEs)
  - File System plug-ins
- Mac OS X KEXT ≠ Mac OS 9 Extension
- Only if you have to…

Kernel Extensions

- Mac OS X Kernel plug-in model
- KEXTs are bundles that contain:
  - Property List in xml
  - A binary for use in the kernel (KMOD)
- KEXTManager auto loads/unloads
- Manual tools, too
  - Kextload, kmodload, knodunload, kmodstat

Kernel Extensions

- Each KMOD has two entry points
  - Initialize, called on load
  - Finalize, called just before unload
- All other APIs are domain specific
  - I/O Kit Drivers
  - Network Kernel Extensions (NKEs)
  - File System Plug-ins

Developing Inside the Kernel

- Two-machine debugging
- Limited language features
- No direct user interaction
- Resources are more costly
- Failures are fatal

Darwin and Open Source

- Virtually all Core OS components are available as open source
- Code is fed back to up-stream providers
  - FreeBSD, Apache, etc.
- Cooperative community
  - Security fixes, platform fixes
- External developers can contribute to Darwin

Roadmap

115 Open Source at Apple	Room A2 **Wed., 10:30am**
Darwin BOF	Hall 2 **Tues., 6:30pm**

Final Things To Think About…

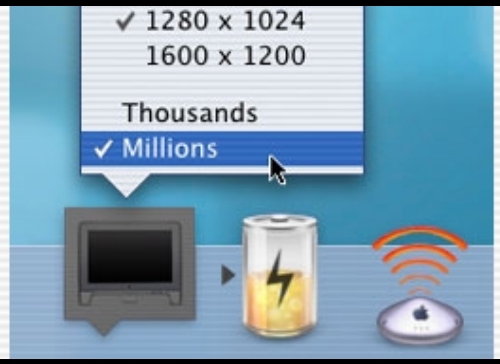
- BSD "UNIX" user experience hidden
  - Optional environment for power users and developers
- Don't repeat bad habits (e.g., shared memory)
- You need to have a "systems" view
- Use highest level of abstraction possible
- Open source is a powerful tool
  - Get involved in Darwin open source!
- There is enormous potential here…

Roadmap

FF121 Feedback Forum: Darwin

Civic
Fri., 5:00pm

Q&A



Jason Yeo
Core OS Technology Manager
jason@apple.com

http://developer.apple.com/events/urls.html

WWDC

Worldwide Developers Conference 2001